In the Claims

This listing of claims will replace all prior versions and listings of claims in this application.

1 (currently amended) A method for improving routing operations in dynamic routing tables comprising:

(a) establishing s to be used in a prefix partitioning scheme, where s denotes a pre-established value of bits of the prefixes;

(b) partitioning at least one prefix into up to min(n, 2^s+ 1) partitions, where n is the total number of prefixes;

(c) assessing prefix length with regard to s;

(d) storing prefixes whose length is larger than or equal to s in a partition that corresponds to the value of the prefix's first s bit;

(e) storing prefixes whose length is smaller than s in a first designated partition;

(f) representing the prefixes in each partition using a router table data structure; and

(g) performing an operation selected from lookup, insert, and delete by using the stored prefixes.

2 (original). The method according to claim 1, wherein the representative router table structure is a dynamic router-table data structure.

3 (original). The method according to claim 2, wherein the dynamic router-table data structure is selected from a group consisting of BOB; PBOB; LMPBOB; B-tree data structure; CRBT; ACRBT; PST; HOT; BOT, and one bit TRIE.

4 (original). The method according to claim 1, wherein the representative router table structure is a static router-table data structure.

5 (original). The method according to claim 4, wherein the static router-table data structure is selected from a group consisting of a linear array; trie-based data structures; and hash trees organized by prefix length.

6 (currently amended). The method according to claim 1, further comprising the steps of:

(a) establishing t to be used for further partitioning the prefixes whose length is smaller than s, where t denotes a second pre-established value of bits of the prefixes;

(b) assessing the length of the prefixes stored in the designated partition with regard to t;

(c) partitioning at least one of the prefixes whose length is larger than or equal to t in up to min (n, $2^t + 1$) partitions, wherein the prefix whose length is smaller than s and larger than or equal to t is placed in a partition based on the value of the prefix's first t bit; and

(d) storing the prefixes whose length is smaller than s and t into a second designated partition.

7 (original). The method according to claim 1, further comprising the step of indexing nonempty partitions using an array or a hash table.

8 (currently amended). A computer program product recorded on computer readable medium for routing packets comprising[[;]]: a computer readable medium for receiving packets specifying prefixes and decision rules for the prefixes; a computer readable medium for establishing s bits to be used in a prefix partitioning scheme; a computer readable medium for matching, inserting, or deleting prefixes in a partitioning tree; and a computer readable medium for performing steps of multilevel partitioning, said steps comprising:

(a) establishing s to be used in a prefix partitioning scheme, where s denotes a pre-established value of bits of the prefixes;

(b) partitioning at least one prefix into up to min (n, $2^s + 1$) partitions, where n is the total number of prefixes;

(c) assessing prefix length with regard to *s*;

(d) storing prefixes whose length is larger than *s* in a partition that corresponds to the value of the prefix's first *s* bit;

(e) storing prefixes whose length is smaller than *s* in a first designated partition; and

(f) representing the prefixes in each partition using a router table data structure.

9 (original). The computer program according to claim 8, wherein the dynamic router-table data structure is selected from a group consisting of BOB; PBOB; LMPBOB; B-tree data structure; CRBT; ACRBT; PST; HOT; and BOT.

10 (original). The computer program according to claim 8, wherein the representative router-table structure is a static router-table data structure.

11 (original). The computer program according to claim 10, wherein the static router-table data structure is selected from a group consisting of a linear array; trie-based data structures; and hash trees organized by prefix length.

12 (currently amended). The computer program according to claim 8, wherein the computer readable medium for performing the steps of multilevel partitioning further comprises the steps of:

(a) establishing *t* to be used for further partitioning the prefixes whose length is smaller than *s*, where *t* denotes a second pre-established value of bits of the prefixes;

(b) assessing the length of the prefixes stored in the designated partition with regard to *t*;

(c) partitioning at least one of the prefixes whose length is larger than or equal to *t* in up to min (n, $2^t + 1$) partitions, wherein the prefix whose length is smaller than *s* and greater than or equal to *t*, is placed in a partition based on the value of the prefix's first *t* bit; and

(d) storing the prefixes whose length is smaller than *s* and *t* into additional bits into a second designated partition.

13 (currently amended). A computer system comprising: a memory containing a partitioning routing table constructing program having functions for constructing a routing table with partitions based on the first *s* bits of a prefix, where *s* denotes a pre-established value of bits of the prefix; wherein the prefixes in each partition are represented by a router table data structure; and a processor for executing the partitioning routing table constructing program.

14 (currently amended). The computer system according to claim 13, wherein the partitioning routing table constructing program further includes functions for constructing partitions based on the first *t* bits of a prefix, where *t* denotes a second pre-established value of bits of the prefix.